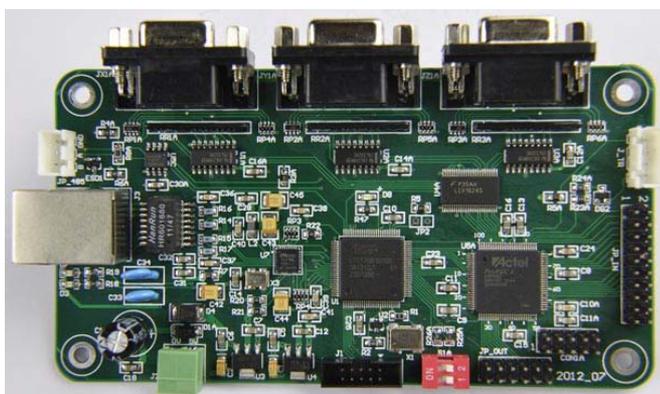


ECC3910 以太网总线三轴光栅尺/ 编码器采集卡



用户开发手册

Version 1.1

上海鑫天精密仪器有限公司

电话：021-54321756

传真：021-54132968

网站：www.sh-xintian.com 邮箱：shxintian@126.com

地址：上海市嘉定区绿苑路488号

目录

一、概述.....	5
二、接口定义及使用	6
1. RJ-45 网络接口	6
2. 拨码开关.....	7
3. 光栅尺接口（JX1A, JY1Z, JZ1A）	7
4. 探针接口(J_TR)	8
5. RS-485 接口(J_11)	8
6. 通用输入接口(J_INPUT).....	9
7. 通用输入接口(J_OUT)	9
8. 电源接口(J2)	9
三、以太网（Ethernet）IP 地址设置	10
1. 出厂默认设置.....	10
2. 用户自定义 IP 地址	10
3. 分布式多卡采集.....	10
四、测试软件使用	10
1. 读数.....	11
2. 设置读值数.....	12
3. 寻找参考点.....	12
4. 设置探针参数.....	12
5. 读锁存值.....	12
6. 通用 I/O 口	13
7. 更改采集卡 IP 地址	13
五、函数库及其使用详细说明	14
函数功能详解：	14
1. 采集卡初始化函数.....	14
2. 关闭采集卡函数.....	15
3. 读三轴编码器计数值函数.....	15
4. 设置轴编码器计数值函数.....	15
5. 读三轴锁存器计数值函数.....	15
6. 设置 EZ（参考点）信号的有效逻辑电平	15
7. 设置探针信号的有效逻辑电平	16
8. 读锁存器触发状态.....	16
9. 复位（恢复）锁存状态，允许重新锁存数据.....	17
10. 某轴搜寻参考点.....	17
11. 三轴同时搜寻参考点.....	17

12.设置探针去抖时间.....	17
13.设置 LED 发光方式.....	17
14.设置光源亮度值.....	18
15.读取光源亮度值.....	18
16.按位写通用输出接口.....	18
17.写通用输出口，一次性修改所有通用输出口.....	18
18.读通用输出口状态值.....	18
19.读通用输入口的电平状态.....	19
20.更改采集卡的 IP 信息.....	19
21.复位采集卡.....	19
六、RS-485 通讯函数.....	19
1. 读三轴编码器计数值函数.....	21
2. 设置编码器值函数.....	21
3. 指定轴寻找参考点函数.....	21
4. 设置参考点信号(EZ)有效电平函数.....	21
5. 读锁存状态和参考点状态函数.....	22
6. 复位锁存器函数，允许重新锁存数据.....	22
7. 读三轴编码器锁存值函数.....	22
8. 设置探针有效动作函数.....	23
9. 设置探针去抖时间函数.....	23
10. 设置探针 LED 状态函数.....	23
11. 按位写通用输出口函数.....	23
12. 按字节写通用输出口函数.....	23
13. 读输出口状态函数.....	24
14. 读输入口状态函数.....	24
15. 设置光源值函数.....	24
16. 读光源值函数.....	24
17. 读多数值函数，此命令可以一次读多个状态数据.....	24
18. 初始化采集卡函数.....	25
19. 设置波特率函数.....	25
20. 打开串口函数.....	25
21. 关闭串口函数.....	25
七、I/O 功能.....	26
八、ECC3910 采集卡机械尺寸图.....	29

一、概述

介绍:

ECC3910 是一款新型光栅尺或旋转编码器等工业测量元件采集卡,用户可以在 PC 机上通过以太网口读取光栅尺或编码器信号,使位置、速度,加速度等测量变得简便,广泛应用于影像仪、测量机等自动化控制设备。

特点:

- ◇ 10M/100M 以太网接口, TCP/IP 协议, 不需要驱动程序
- ◇ 3 轴光栅尺或编码器信号输入
- ◇ 输入信号单端输入或者差分输入均可
- ◇ 硬件计数, 最大输入频率 4MHz
- ◇ 一路探针触发信号, 硬件锁存数据
- ◇ 通用 IO 口, 8 输出, 16 输入
- ◇ 保留 RS-485 接口, 方便集成到 PLC 系统中
- ◇ 提供 API 函数和演示程序, 方便二次开发
- ◇ 通道光源控制, 每通道光亮等级 200 级。(仅 ECC3910L 型号)

软件支持:

适用于 Windows、linux 或嵌入式系统, 支持 Visual C++6.0, Borland C++ Builder, Visual Basic 6.0 等软件开发平台。

技术规格:

编码器信号

编码器轴数	3
计数器位数	32 位
编码器信号类型	A、B 相正交信号
计数频率	4MHz
提供各编码器电源	5V±5%, 最大 100mA

数字 I/O 信号

输入口数	16
输出口数	8
I/O 信号类型	3.3V TTL 电平

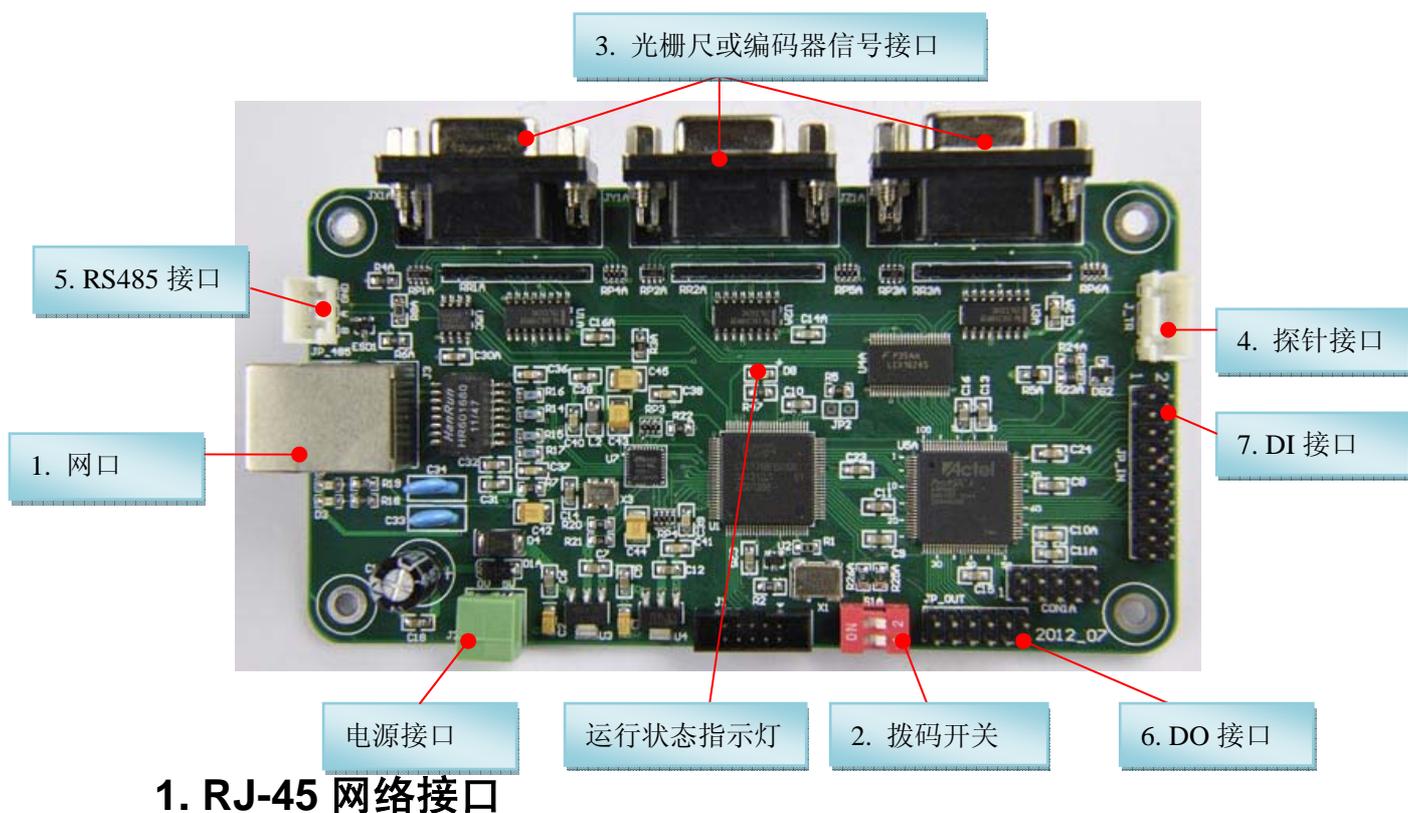
常规指标

电源	5V, 最大 500mA
操作系统	Windows XP、2K 或 Win7
温度范围	0℃~50℃
湿度范围	5~85%, 非结露
尺寸	130mm(长)*70mm(宽)

ECC系列采集卡选型表:

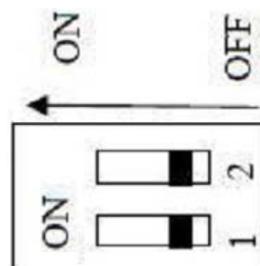
订货型号	通讯方式	轴数	是否支持多卡?	其它通讯方式	DI (数字输入)	DO (数字输出)
ECC3910	USB 总线	3	否	RS-232	16 个	8 个
ECC3910	以太网	3	是	RS-485	16 个	8 个

二、接口定义及使用



2. 拨码开关

ECC3910 采集卡上有一只两位的拨码开关，用来设定采集卡上电后数字输出点 (D0) 的初始电平。上电后改变拨码位状态不会改变输出口电平。



输出口	拨码位编号	ON	OFF
OUT1~OUT4	1	高电平	低电平
OUT5~OUT8	2	高电平	低电平

3. 光栅尺接口 (JX1A, JY1Z, JZ1A)

DB9九针插接头

	TTL		422A	
	1	2	3	4
1			A-	
2	0V		0V	
3			B-	
4				
5			R-	
6	A		A+	
7	5V		5V	
8	B		B+	
9	R		R+	

DB9 针插头 (用于编码器或光栅尺), 信和、新天, 长春 TLL 或 RS-422 电平光栅尺可直接接入。

万濠、七海光电光栅尺请参考《采集卡用户开发手册》中的连接对照表。

品牌	万濠				DB9 差分尺					
引脚编号	1	2	3	4	5	6	7	8	9	SHELL
引脚定义	VCC	0V	A+	B+	R+	R-	A-	B-	SHELL	

品牌	万濠				DB9 单端尺					
引脚编号	1	2	3	4	5	6	7	8	9	SHELL
引脚定义	VCC	0V	A+	B+	R+	/	/	/	SHELL	

品牌	七海				DB9 单端尺					
引脚编号	1	2	3	4	5	6	7	8	9	SHELL
引脚定义	VCC	0V	A+	B+	R+	/	/	/	/	

品牌	信和或新天				DB9 差分尺					
引脚编号	1	2	3	4	5	6	7	8	9	SHELL
引脚定义	A-	0V	B-	SHELL	R-	A+	VCC	B+	R+	

品牌	信和或新天				DB9 单端尺					
引脚编号	1	2	3	4	5	6	7	8	9	SHELL
引脚定义	/	0V	/	SHELL	/	A+	VCC	B+	R+	

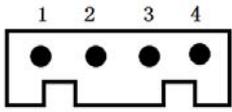
表 2-1 各品牌光栅尺接口定义

RENISHAW 光栅尺与 ECC3910 (L) 采集卡接线表:

ECC3910 (L) DB9 接口编号	ECC3910 (L) DB9 接口定义	RENISHAW 光栅尺 DB15 接口编号	RENISHAW 光栅尺 DB15 接口定义
1	A-	6	A-
2	0V	2&9 (注 1)	0V
3	B-	5	B-
4	NC		
5	R-	4	R-
6	A+	14	A+
7	VCC	7&8 (注 1)	5V
8	B+	13	B+
9	R+	12	R+
SHELL	SHELL	SHELL	SHELL

注 1: 两个脚的线必须同时接到 ECC3910 (L) 的相应引脚上。

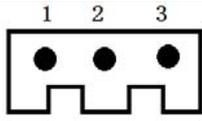
4. 探针接口(J_TR)

探针接口(J_TR)		
引脚编号	定义	端子图
1	TR, 探针正	
2	GND, 探针负	
3	LED+, 探针 LED+	
4	GND, 探针 LED-	

注: 端子座焊盘方形的为第一脚。

5. RS-485 接口(J_11)

RS-485 接口(J_11)		
引脚编号	定义	端子图

1	B	
2	A	
3	GND, 0V	

注：端子座焊盘方形的为第一脚。

6. 通用输入接口(J_INPUT)

J_INPUT 是通用输入接口，IDC 型 20 针插针，定义如下：

脚号	I/O	功能	脚号	I/O	功能
1	I	通用输入接口 1	11	I	通用输入接口 11
2	I	通用输入接口 2	12	I	通用输入接口 12
3	I	通用输入接口 3	13	I	通用输入接口 13
4	I	通用输入接口 4	14	I	通用输入接口 14
5	I	通用输入接口 5	15	I	通用输入接口 15
6	I	通用输入接口 6	16	I	通用输入接口 16
7	I	通用输入接口 7	17		GND
8	I	通用输入接口 8	18		GND
9	I	通用输入接口 9	19		3.3V
10	I	通用输入接口 10	20		3.3V

7. 通用输出接口(J_OUT)

J_OUT 是通用输出接口，IDC 型 12 针插针，定义如下：

脚号	I/O	功能	脚号	I/O	功能
1	0	通用输出接口 1	7	0	通用输出接口 7
2	0	通用输出接口 2	8	0	通用输出接口 8
3	0	通用输出接口 3	9		GND
4	0	通用输出接口 4	10		GND
5	0	通用输出接口 5	11		3.3V
6	0	通用输出接口 6	12		3.3V

注：1. 板上有一个两位的拨码开关，可用来设置采集卡上电后输出口状态。详见本章第 2 节描述。第 1 位为 ON 时，通用输出口 1~4 上电输出高电平，为 OFF 时通用输出口 1~4 输出低电平。；第 2 位为 ON 时，通用输出口 5~8 输出高电平；为 OFF 时通用输出口 5~8 输出低电平。

8. 电源接口(J2)

脚号	定义	脚号	定义
1	电源, 5V	2	地, 0V

注：端子座焊盘方形的为第一脚。

三、以太网（Ethernet）IP 地址设置

1. 出厂默认设置

- 1.1 IP 地址： 192.168.1.100
- 1.2 子网掩码： 255.255.255.0
- 1.3 默认网关： 192.168.1.1

2. 用户自定义 IP 地址

用户可以自定义采集卡的 IP 地址，子网掩码和网关（请参考第四章第 7 节和第五章第 20 节）。当配置错误或者忘记自定义的 IP 地址时，可通过如下步骤恢复出厂设置：

第一步：先让 ECC3910 采集卡处于上电状态，然后短接 JP2（位于板的中心位置附近），保持至少 3 秒钟。

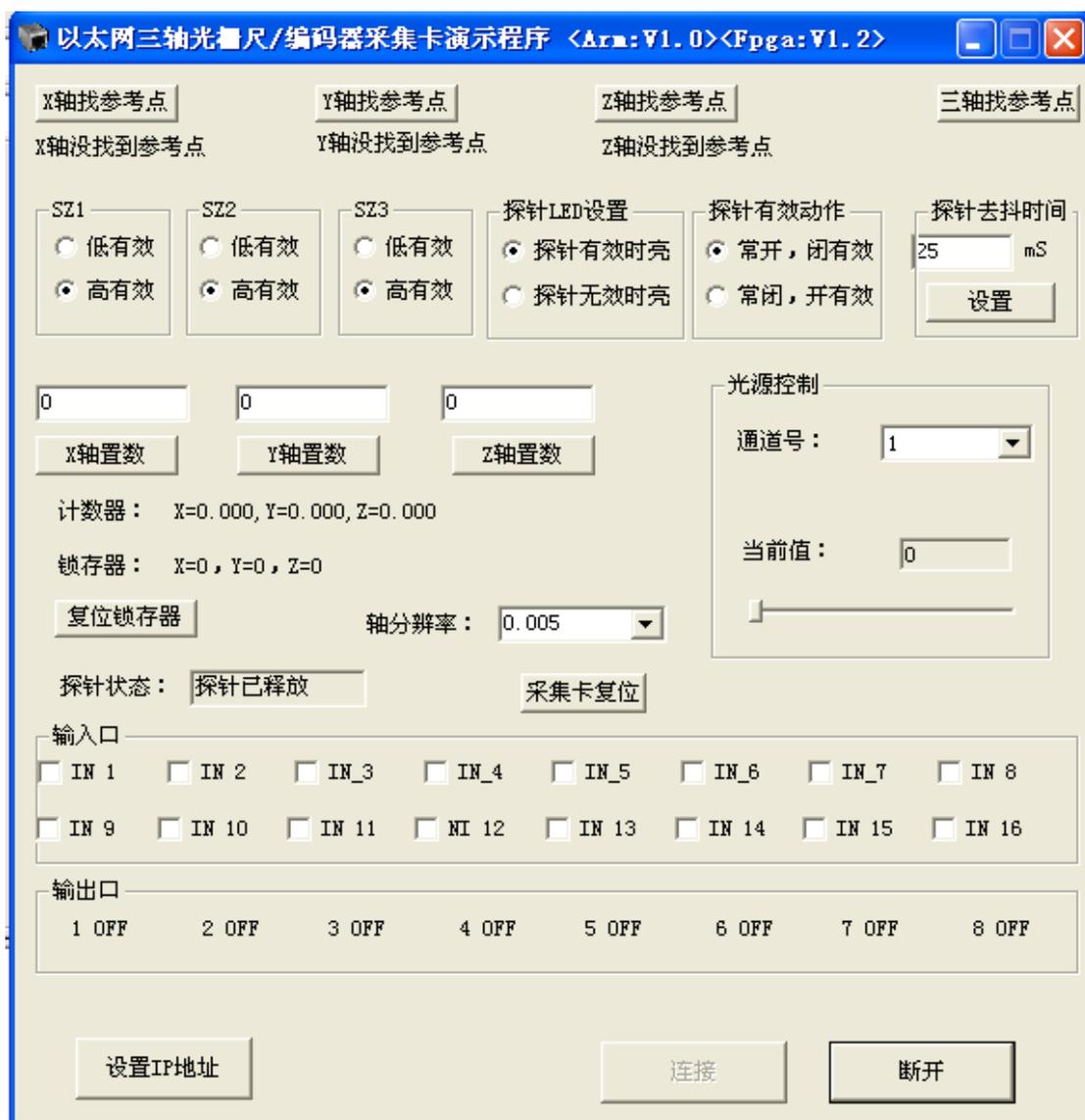
第二步：关闭电源，不要短接 JP2，上电运行，这时 ECC3910 采集卡恢复到出厂时默认的 IP 地址。

3. 分布式多卡采集

ECC3910 是以太网总线式采集卡，用户先给每张采集卡分配不同的 IP 地址，通过集线器扩展网口，可支持多卡同时采集。

四、测试软件使用

Net_Demo.exe 是配套采集的演示程序，运行后整个界面如下：



注：连接成功后，软件顶部的标题栏会显示采集卡的固件版本号。

1. 读数

```
计数器： X=0.005, Y=-12.585, Z=0.000
锁存器： X=0.000, Y=80.625, Z=0.000
```

计数器显示三轴编码器的读数。锁存器显示的是探针动作时锁存到的三轴数据。读编码器函数和读锁存器值函数请参考表 4-1 的函数 1 和函数 5。

2. 设置读数值

单独对某轴设置读数值，设置后的读数会读到新的设置数。设置编码器值函数请参考表 4-1 的函数 4。

3. 寻找参考点

可以单独对某轴或同时对三轴寻找光栅尺或编码器的参考点，在**寻找参考点**功能开启的前提下，第一次碰到参考点信号，采集卡相应轴的读数会清零。如果想再次寻找参考点，可以再开启**寻找参考点**功能。关于寻找参考点的函数请参考表 4-1 的函数 8、函数 10 和函数 11。

4. 设置探针参数

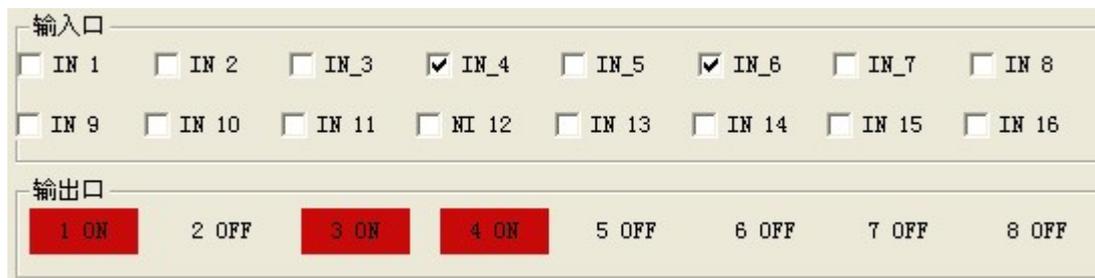
探针默认设置时间 25mS，如果是保持时间比较小的触发信号当探针信号，请根据实际情况修改去抖时间。关于探针设置的函数请参考函数 7，函数 12 和函数 13。

当探针触发信号由外部芯片引起时，可能不需要去抖动时间，可以将探针去抖时间设置为 0。

5. 读锁存值

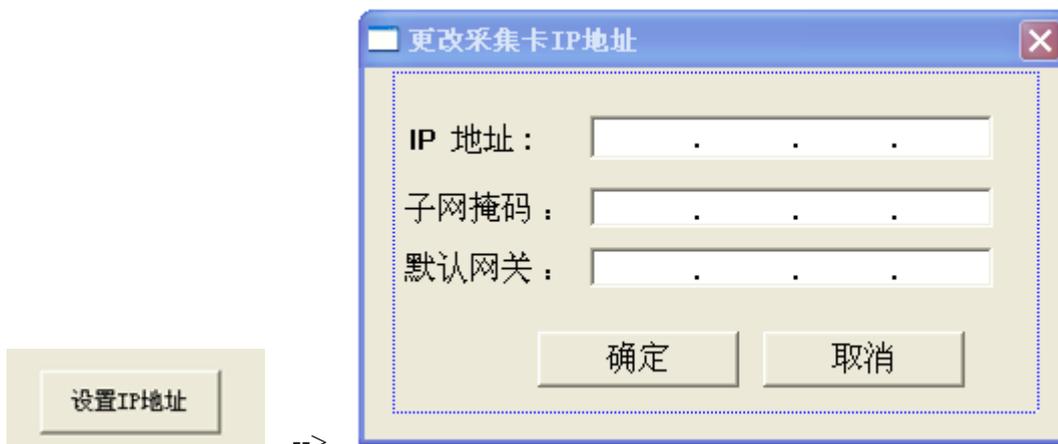
在允许锁存的前提下，探针有效信号产生后，硬件会把当时三轴的数据锁存起来，用户可以根据函数 8 返回的值判断是否有锁存标志产生。一旦锁存数据后，通过函数 5 可以读得锁存值。复位锁存器才能进行下一次锁存。

6. 通用 I/O 口



输入高电平时，对应输入口有“√”号出现。点击输出口，红色表示输出高电平，灰色表示输出低电平。关于通用 I/O 口的函数可参考函数 16 至函数 19。

7. 更改采集卡 IP 地址



连接成功后，可点击“设置 IP 地址”按钮更改采集卡的 IP 地址，更改后的 IP 信息将会保存到软件运行目录下的 ipconfig.ini 文件中，下次运行将会从此文件读取 IP。

注：

1. 在更改 IP 成功后，采集卡要重新上电才会按新的 IP 地址运行。
2. 如果连接失败请检查采集卡的 IP 与 ipconfig.ini 文件中的 IP 信息是否一致。
3. 如果以上都不可行，请参考第三章第 2 节将采集卡的 IP 信息设置为默认值，并把 ipconfig.ini 文件删除，再修改 PC 的本地连接 IP 信息使得与采集卡 IP 一致。

五、函数库及其使用详细说明

序号	函数名	功能描述
1	BOOL Net_Init(const char* addr)	卡初始化函数
2	void Net_Close(void)	关闭卡
3	BOOL Net_Get_Encoder(long* value)	读三轴编码器值
4	BOOL Net_Set_Encoder(char axis, long value)	设置编码器值
5	BOOL Net_Get_Latch_Value(long* value)	读三轴锁存器的值
6	BOOL Net_Set_EZ_Logic(char axis, char logic)	设置 EZ 信号有效电平, 默认是低电平有效
7	BOOL Net_Set_Triger_Logic(char logic)	设置探针有效电平, 默认是低电平有效
8	BOOL Net_Read_Latch_Status(long* status)	读锁存器状态和参考点状态
9	BOOL Net_Reset_Latch_Flag(void)	复位锁存器
10	BOOL Net_Search_Axis_Ref(char axis)	指定轴找参考点功能
11	BOOL Net_Search_All_Ref(void)	所有轴找参考点功能
12	BOOL Net_Set_Triger_Time(short value)	设置探针去抖时间, 默认 25ms
13	BOOL Net_Set_Led (char logic)	设置 LED 发光方式
14	BOOL Net_Set_Light (unsigned char* light)	设置光源 (ECC3910L 可用)
15	BOOL Net_Read_Light (unsigned char* light)	读光源值 (ECC3910L 可用)
16	BOOL Net_Write_OutBit(unsigned char bitno , unsigned char logic);	按位写通用输出口
17	BOOL Net_Write_OutPort(unsigned char value);	写通用输出口
18	BOOL Net_Read_OutPort(unsigned char* value);	读通用输出口
19	BOOL Net_Read_InPort(unsigned short* value);	读通用输入口
20	BOOL Net_Change_IP(unsigned char* ip, unsigned char* mark, unsigned char* gateway);	更改采集卡 IP 信息
21	BOOL Net_ResetCard(void);	复位采集卡

表 5-1 函数列表

函数功能详解:

1.采集卡初始化函数

函数: **BOOL __stdcall Net_Init(const char* addr)**

参数: addr 为采集卡的 IP 地址, 如:

```
addr[4] = {192, 168, 1, 100};
```

返回值：1 – 成功；0 – 失败。

2.关闭采集卡函数

函数：**void __stdcall Net_Close(void)**

参数：无。

返回值：无。

3.读三轴编码器计数值函数

函数：**BOOL __stdcall Net_Get_Encoder(long* value)**

参数：指向 long 型指针，用来返回三轴读数。

返回值：1 – 成功；0 – 失败。

例程：`long value[3];`

`Net_Get_Encoder(value);`//调用成功后，value 数组有 0~2 元素分别保存 0~2 轴数值。

4.设置轴编码器计数值函数

函数：**BOOL __stdcall Net_Set_Encoder(char axis, long value)**

参数：axis 表示轴号

value 指定计数初始值。

返回值：1 – 成功；0 – 失败。

例程：

`Net_Set_Encoder(0,1000);`//将 0 轴编码器当前计数值设为 1000。

5.读三轴锁存器计数值函数

函数：**BOOL __stdcall Net_Get_Latch_Value(long* value)**

参数：指向 long 型指针，用来返回四轴值。

返回值：1 – 成功；0 – 失败。

例程：`long value[6];`//注意数组长度要大于等于 6

`Net_Get_Encoder(value);`//调用成功后，value 数组 0~2 元素分别保存 0~2 轴锁存值。

6.设置 EZ（参考点）信号的有效逻辑电平

函数：**BOOL __stdcall Net_Set_EZ_Logic(char axis, char logic)**

参数：axis 表示轴号：0、1、2；logic 定义 EZ 信号的有效逻辑电平：0-低电平

1-高电平。ECC3910 卡上电默认为高电平有效。

返回值：1 – 成功；0 – 失败。

注：EZ 为编码器的第 3 相信号，一般用做回零用。对于旋转编码器命名为：EZ+ 和 EZ-；对于光栅尺，一般命名为：RI+和 RI-。

7.设置探针信号的有效逻辑电平

函数：**BOOL __stdcall Net_Set_Triger_Logic(char logic)**

参数：logic 定义探针的有效动作：0-常开，闭合有效，1-常闭，断开有效。
ECC3910 卡上电默认为常开，闭合有效。

返回值：1 – 成功；0 – 失败。

8.读锁存器触发状态

函数：**BOOL __stdcall Net_Read_Latch_Status(long* status)**

参数：指向 long 型指针，用来返回锁存及寻找参考点状态。

返回值：见下表

返回值位号	描述
0~2	分别代表 0~2 轴的触发状态：0-复位状态（无触发），1-触发状态（表示已经产生了触发锁存）； 注：任何轴的触发状态值为 1 时，即使外触发信号产生并满足规定的跳变边沿，这个轴的锁存器也不会锁存新的计数值，直到调用 Net_Reset_Latch_Flag(void)函数复位触发状态。 BIT0---X 轴；BIT1---Y 轴；BIT2---Z 轴
3	保留
4	探针状态位：0-复位状态（探针已经释放），1-触发状态（探针未释放）
5~7	保留
8~10	分别代表 0~2 轴正在寻找参考点状态：0-复位状态（没进行寻找参考点），1-触发状态（正在找参考点）；调用 Net_Search_Axis_Ref() 成功返回后，对应轴的位会置 1。 BIT8---X 轴；BIT9---Y 轴；BIT10---Z 轴
11	保留
12~14	分别代表 0~2 轴已寻到参考点状态：0-复位状态（未找到参考点），1-触发状态（表示已找到参考点）。 BIT12---X 轴；BIT13---Y 轴；BIT14---Z 轴
15	保留
16~18	分别代表 0~2 轴的 EZ(参考点)信号的电平：0-低电平；1-高电平
19~31	保留

9.复位（恢复）锁存状态，允许重新锁存数据

函数：**BOOL __stdcall Net_Reset_Latch_Flag(void)**

参数：无。

返回值：1 – 成功；0 – 失败。

10.某轴搜寻参考点

函数名：**BOOL __stdcall Net_Search_Axis_Ref(char axis)**

参数：axis 表示轴号：0、1、2；

返回值：1 – 成功；0 – 失败。

注意：轴第一次碰到 EZ（参考点）信号将该轴的读数清 0。

11.三轴同时搜寻参考点

函数名：**BOOL __stdcall Net_Search_All_Ref(void)**

参数：无；

返回值：1 – 成功；0 – 失败。

注意：每轴第一次碰到 EZ（参考点）信号将该轴的读数清 0。

12.设置探针去抖时间

函数名：**BOOL __stdcall Net_Set_Triger_Time(unsigned short value)**

参数：value 指定去抖时间，采集卡上电默认为 25ms。

返回值：1 – 成功；0 – 失败。

13.设置 LED 发光方式

函数名：**BOOL __stdcall Net_Set_Led(char logic)**

参数：logic 定义 LED 发光方式。

logic = 0，探针触发有效时 LED 亮，探针无效 LED 灭（缺省设置）；logic = 1，探针触发有效时 LED 灭，探针无效时 LED 亮。

返回值：1 – 成功；0 – 失败。

14.设置光源亮度值

函数名: **BOOL __stdcall Net_Set_Light (unsigned char* light)**

参数: Light[0]- Light[4]为 5 路光源亮度, 0~200 级别。

返回值: 1 – 成功; 0 – 失败。

15.读取光源亮度值

函数名: **BOOL __stdcall Net_Read_Light (unsigned char* light)**

参数: Light[0]- Light[4]为采集卡返回的 5 路光源亮度值。

返回值: 1 – 成功; 0 – 失败。

16.按位写通用输出接口

函数名: **BOOL __stdcall Net_Write_OutBit (unsigned char bitno, unsigned char logic)**

参数: bitno 表示通用输出口位号;

logic 表示输出状态, 0—低电平; 1—高电平。

返回值: 1 – 成功; 0 – 失败。

注意: ECC3910 有 8 位输出口, 位编号按 1~8 标定。

17.写通用输出口, 一次性修改所有通用输出口

函数名: **BOOL __stdcall Net_Write_OutPort (unsigned char value)**

参数: value 为通用输出口控制字, 0~7 位分别表示通用输出口 1~8 位的输出状态。

返回值: 1 – 成功; 0 – 失败。

18.读通用输出口状态值

函数名: **BOOL __stdcall Net_Read_OutPort (unsigned char* value)**

参数: value 为指针, 用作返回通用输出口状态字, 0~7 位分别表示通用输出口 1~8 位的输出状态。

返回值: 1 – 成功; 0 – 失败。

19.读通用输入口的电平状态

函数名: `BOOL __stdcall Net_Read_InPort (unsigned short* value)`

参数: 指向 short 型指针，用来返回 16 位通用输入口状态。返回值的 0~15 位数据分别表示通用输入口的输入位 1~16 位。输入口为电平时，对应位为 0，否则为 1。

返回值: 1 – 成功；0 – 失败

注意: ECC3910 共有 16 位的通用输入口，对应位号是 1~16。

20.更改采集卡的 IP 信息

函数名: `BOOL __stdcall Net_Change_IP(unsigned char* ip, unsigned char* mark, unsigned char* gateway)`

参数: ip、mark、gateway 分别为要更改的 IP 地址、子网掩码、网关。如：

```
ip[4] = {192, 168, 2, 100};
```

```
mark[4] = {255, 255, 255, 0};
```

```
gateway[4] = {192, 168, 2, 1};
```

返回值: 1 – 成功；0 – 失败

21.复位采集卡

函数名: `BOOL __stdcall Net_ResetCard(void)`

参数: 无

返回值: 1 – 成功；0 – 失败

注意: 调用该函数后参数恢复上电初的值。

六、RS-485 通讯函数

ECC3910 采集卡保留 RS-485，方便用户集成到自己的单片机板、PLC 或嵌入式系统中。如果用户不需要用到 RS-485 接口，可不完全理会这章节。

6.1 通讯参数

波特率	上电默认 38400BPS
数据位	8 位
停止位	1 位
校验	无校验

6.2 RS-485 函数说明

RS-485 通讯函数和以太网通讯函数比较类似，都包含在同一库文件中。用户只需调用函数可以完成 RS-845 通讯功能。若用户主机是 PLC 或其他嵌入式芯片，可以与我司销售人员联系取得底层的通讯协议。

由于函数封装到 DLL 文件中，用户需调用第 20 号函数打开串口后，才能调用其它函数进行通讯，程序退出则调用第 21 号函数关闭串口。

为了增加函数的可靠性，调用函数 100mS 后，若未收到采集卡响应，函数则返回超时错误。

6.3 RS-485 函数列表

序号	函数名	功能描述
1	int RS485_Get_Encoder(long* value)	读三轴编码器值
2	int RS485_Set_Encoder(char axis, long value)	设置编码器值
3	int RS485_Search_Axis_Ref(char axis)	指定轴找参考点功能
4	int RS485_Set_EZ_Logic (unsigned char x, unsigned char y, unsigned char z)	设置 EZ 信号有效电平，默认是低电平有效
5	int RS485_Read_Latch_Status(long* status)	读锁存器状态和参考点状态
6	int RS485_Reset_Latch_Flag(void)	复位锁存器
7	int RS485_Get_Latch_Value(long* value)	读三轴锁存器的值
8	int RS485_Set_Triger_Logic(char logic)	设置探针有效电平，默认是低电平有效
9	int RS485_Set_Triger_Time (unsigned short value)	设置探针去抖时间，默认 25ms
10	int RS485_Set_Led(char logic)	设置 LED 发光方式
11	int RS485_Write_OutBit (unsigned char bitno, unsigned char logic)	按位写通用输出口
12	int RS485_Write_OutPort (unsigned char value)	写通用输出口
13	int RS485_Read_OutPort (unsigned char* value)	读通用输出口
14	int RS485_Read_InPort (unsigned short* value)	读通用输入口
15	int RS485_Set_Light (unsigned char* light)	设置光源 (ECC3910L 可用)
16	int RS485_Read_Light (unsigned char* light)	读光源值 (ECC3910L 可用)
17	int RS485_Read_Multi (long* latchstatus, unsigned short* input, unsigned char* ourport, unsigned char* light)	读多个数值
18	int RS485_Init_Card (void)	复位采集卡
19	int RS485_Set_Baud(unsigned char baud)	设置波特率
20	int RS485_ComOpen(char* port, unsigned long BaudRate, unsigned char ByteSize, unsigned char StopBits)	打开串口函数

21	int RS485_ComClose ()	关闭串口函数
----	-----------------------	--------

表 6-3 函数列表

6.4 RS-485 函数使用详解

1. 读三轴编码器计数值函数

函数: int __stdcall RS485_Get_Encoder(long* value)

参数: 指向 long 型指针, 用来返回三轴读数值。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

例程: long value[3];

```
RS485_Get_Encoder(value); //调用成功后, value 数组有 0~2 元素分别保//
存 0~2 轴数值。
```

2. 设置编码器值函数

函数: int __stdcall RS485_Set_Encoder(char axis, long value)

参数: axis 表示轴号

value 指定计数初始值。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

例程:

```
RS485_Set_Encoder(0,1000); //将 0 轴编码器当前计数值设为 1000。
```

3. 指定轴寻找参考点函数

函数名: int __stdcall RS485_Search_Axis_Ref(char axis)

参数: axis 表示轴号: 0—X 轴; 1—Y 轴; 2—Z 轴;

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

4. 设置参考点信号(EZ)有效电平函数

函数: int __stdcall RS485_Set_EZ_Logic (unsigned char x, unsigned char y, unsigned char z)

参数: x、y、z 分别代表三轴, 0x00---低电平, 0x01---高电平

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

注: EZ 为编码器的第 3 相信号, 一般用做回零用。对于旋转编码器命名为: EZ+ 和 EZ-; 对于光栅尺, 一般命名为: RI+和 RI-。

5. 读锁存状态和参考点状态函数

函数： `int __stdcall RS485_Read_Latch_Status(long* status)`

参数： 指向 long 型指针，用来返回锁存及寻找参考点状态。

返回值： 1 – 成功； 0 – 发送失败； -1 – 超时； -2 – 校验位出错

说明： 通过指针返回的参数 `status` 定义如下表：

位号 (bit)	描述
0~2	分别代表 0~2 轴的触发状态：0-复位状态（无触发），1-触发状态（表示已经产生了触发锁存）； 注：触发后，直到复位锁存器后（详见第 6 号命令字），才能锁存新的数据。 BIT0---X 轴； BIT1---Y 轴； BIT2---Z 轴
3	保留
4	探针状态位：0-复位状态（探针已经释放），1-触发状态（探针未释放）
5~7	保留
8~10	分别代表 0~2 轴正在寻找参考点状态：0-复位状态（没进行寻找参考点），1-触发状态（正在找参考点）；查找参考点（详见第 3 号命令字）成功返回后，对应轴的位会置 1。 BIT8---X 轴； BIT9---Y 轴； BIT10---Z 轴
11	保留
12~14	分别代表 0~2 轴已寻到参考点状态：0-复位状态（未找到参考点），1-触发状态（表示已找到参考点）。 BIT12---X 轴； BIT13---Y 轴； BIT14---Z 轴
15	保留
16~18	分别代表 0~2 轴的 EZ(参考点)信号的电平：0-低电平； 1-高电平
19~31	保留

6. 复位锁存器函数，允许重新锁存数据

函数： `int __stdcall RS485_Reset_Latch_Flag(void)`

参数： 无。

返回值： 1 – 成功； 0 – 发送失败； -1 – 超时； -2 – 校验位出错

7. 读三轴编码器锁存值函数

函数： `int __stdcall RS485_Get_Latch_Value(long* value)`

参数： 指向 long 型指针，用来返回四轴值。

返回值： 1 – 成功； 0 – 发送失败； -1 – 超时； -2 – 校验位出错

例程： `long value[3];`

`RS485_Get_Encoder(value);` //调用成功后, value 数组有 0~2 元素分别

//保存 0~2 轴锁存值。

8. 设置探针有效动作函数

函数名: **int __stdcall RS485_Set_Triger_Logic(char logic)**

参数: **logic** 定义探针的有效动作: 0-常开, 闭合有效; 1-常闭, 断开有效。
ECC3910 卡上电默认为常开, 闭合有效。

返回值: 1- 成功; 0- 发送失败; -1- 超时; -2- 校验位出错

9. 设置探针去抖时间函数

函数名: **int __stdcall RS485_Set_Triger_Time (unsigned short value)**

参数: **value** 指定去抖时间, 采集卡上电默认为 25ms。

返回值: 1- 成功; 0- 发送失败; -1- 超时; -2- 校验位出错

10. 设置探针 LED 状态函数

函数名: **int __stdcall RS485_Set_Led(char logic)**

参数: **logic** 定义 LED 发光方式。

logic = 0, 探针触发有效时 LED 亮, 探针无效 LED 灭 (缺省设置); **logic = 1**, 探针触发有效时 LED 灭, 探针无效时 LED 亮。

返回值: 1- 成功; 0- 发送失败; -1- 超时; -2- 校验位出错

11. 按位写通用输出口函数

函数名: **int __stdcall RS485_Write_OutBit (unsigned char bitno, unsigned char logic)**

参数: **bitno** 表示通用输出口位号;

logic 表示输出状态, 0—低电平; 1—高电平。

返回值: 1- 接收成功; 0- 发送失败; -1- 超时; -2- 校验位出错

注意: ECC3910 有 8 位输出口, 位编号按 1~8 标定。

12. 按字节写通用输出口函数

函数名: **int __stdcall RS485_Write_OutPort (unsigned char value)**

参数: **value** 为通用输出口控制字, 0~7 位分别表示通用输出口 1~8 位的输出状态。

返回值: 1- 成功; 0- 发送失败; -1- 超时; -2- 校验位出错

13. 读输出口状态函数

函数名: **int __stdcall RS485_Read_OutPort (unsigned char* value)**

参数: value 为指针, 用作返回通用输出口状态字, 0~7 位分别表示通用输出口 1~8 位的输出状态。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

14. 读输入口状态函数

函数名: **int __stdcall RS485_Read_InPort (unsigned short* value)**

参数: 指向 short 型指针, 用来返回 16 位通用输入口状态。返回值的 0~15 位数据分别表示通用输入口的输入位 1~16 位。输入口为电平时, 对应位为 0, 否则为 1。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

注意: ECC3910 共有 16 位的通用输入口, 对应位号是 1~16。

15. 设置光源值函数

函数名: **int __stdcall RS485_Set_Light (unsigned char* light)**

参数: Light[0]- Light[4]为 5 路光源亮度, 0~200 级别。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

16. 读光源值函数

函数名: **int __stdcall RS485_Read_Light (unsigned char* light)**

参数: Light[0]- Light[4]为采集卡返回的 5 路光源亮度值。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

17. 读多数值函数, 此命令可以一次读多个状态数据

函数名: **int __stdcall RS485_Read_Multi (long* latchstatus, unsigned short* input, unsigned char* ourport, unsigned char* light)**

参数: latchstatus 表示锁存状态字, 详见第 5 函数;

input 表示输入状态字, 详见第 14 函数;

ourport 表示输出状态字, 详见 13 函数;

light 表示光源亮度值, 详见第 16 函数

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

18. 初始化采集卡函数

函数: `int __stdcall RS485_Init_Card (void)`

参数: 无

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

注: 初始化后, 采集卡恢复到上电默认状态。

19. 设置波特率函数

函数名: `int __stdcall RS485_Set_Baud(unsigned char baud)`

参数: `baud` 可取的值有: 0 到 5。分别代表: 0—4800BPS, 1—9600BPS, 2—19200BPS, 3—38400BPS, 4—57600BPS, 5—115200BPS, 采集卡上的默认波特率为 38400BPS。

返回值: 1 – 成功; 0 – 发送失败; -1 – 超时; -2 – 校验位出错

注: 函数成功调用后, 新的波特率才正式生效。主机要按新的波特率和采集卡通讯。

20. 打开串口函数

函数名: `int __stdcall RS485_ComOpen(char* port, unsigned long BaudRate, unsigned char ByteSize, unsigned char StopBits)`

参数: `port` – 串口号, 用户 PC 端的串口号

`BaudRate` – 波特率, 与采集卡当前波特率一致

`ByteSize` – 数据位, 必须和采集卡匹配, 此参数值固定为 8;

`StopBits` – 停止位, 必须和采集卡匹配, 此参数值固定为 0;

返回值: 1 – 成功; 0 – 失败;

注: 用户必须要调用此函数打开串口后, 才能调用其他通讯函数

例: `RS485_ComOpen(“COM3”, 38400, 8, 0);` //打开串口 COM3, 波特率为 38400, 8 位数据位, 1 位停止位;

21. 关闭串口函数

函数名: `int __stdcall RS485_ComClose ()`

参数: 无

返回值: 1 – 成功; 0 – 失败;

七、I/O 功能

ECC3910 采集卡有 16 路数字输入(DI)和 8 路数字输出(DO), DI 和 DO 均由板上 FPGA 器件扩展出来, 电平是 LVTTTL。由于测量机或工控设备中输入输出多数为 12V 或 24V 器件, 因此建议客户设计接口时使用光耦隔离方式, 亦可订购我司的 IO-1608 光电隔离板。

7.1 IO-1608 光电隔离板原理图

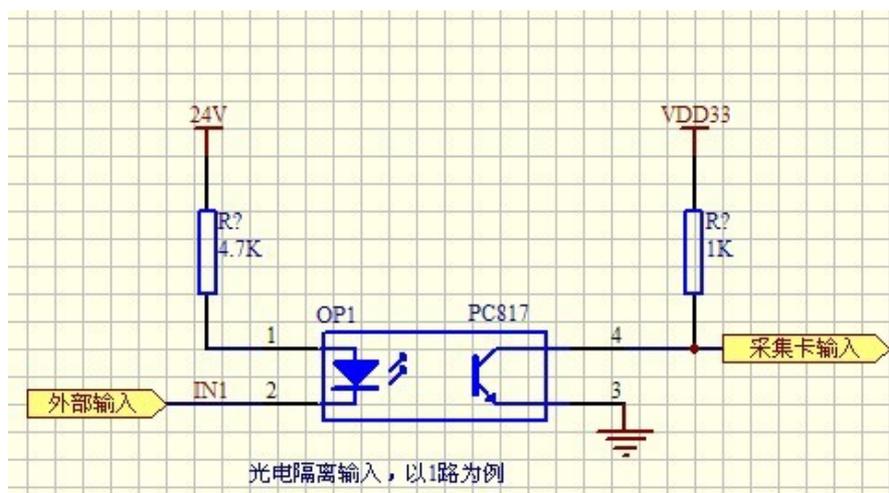


图 7-1 IO-1608 光电隔离板输入原理图

由上图可知, 接上光电隔离板时, ECC3910 采集卡端所有输入的电平状态都是高电平, 外部输入有效时 (接通外部电源地), ECC3910 采集卡对应输入点电平才为低。

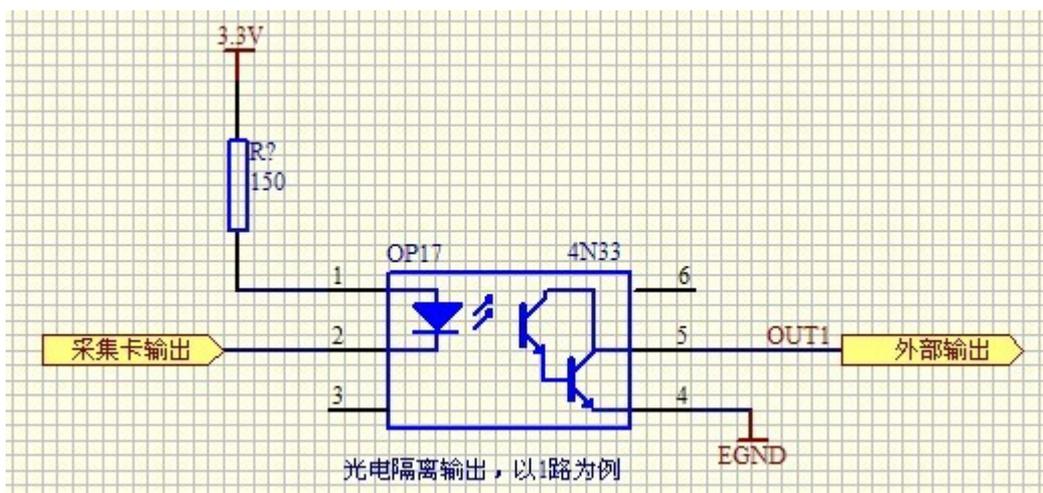


图 7-2 IO-1608 光电隔离板输出原理图

由上图可知, 接上光电隔离板时, ECC3910 采集卡输出端为低电平时, 光耦

4N33 导通，外部输出点和外部电源地接通；ECC3910 采集卡输出端为高电平时，光耦 4N33 不导通，外部输出点和外部电源地断开。用户可以通过拨码开关(详见第二章第 2 节)来设置采集卡输出点上电时的初始电平。

7.2 IO-1608 光电隔离板接线定义

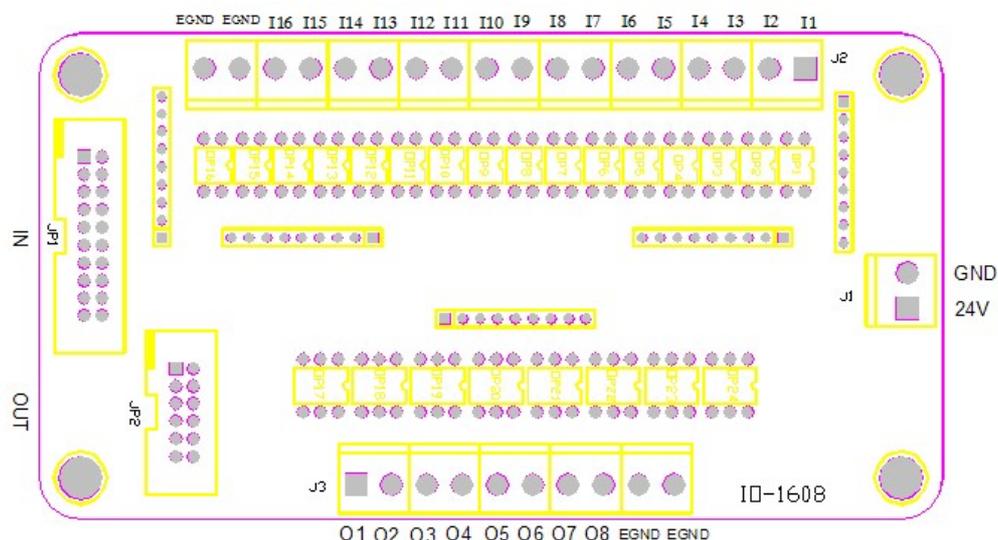


图 7_3 IO-1608 接口定义图

7.2.1 J1---外部电源接口

引脚	名称	功能
1	12V~24V	外部电源+
2	EGND	外部电源参考地

7.2.2 J2---外部输入接口

引脚	名称	I/O	功能	引脚	名称	I/O	功能
1	IN1	I	外部输入 1	10	IN10	I	外部输入 10
2	IN2	I	外部输入 2	11	IN11	I	外部输入 11
3	IN3	I	外部输入 3	12	IN12	I	外部输入 12
4	IN4	I	外部输入 4	13	IN13	I	外部输入 13
5	IN5	I	外部输入 5	14	IN14	I	外部输入 14
6	IN6	I	外部输入 6	15	IN15	I	外部输入 15
7	IN7	I	外部输入 7	16	IN16	I	外部输入 16
8	IN8	I	外部输入 8	17	EGND		外部电源参考地
9	IN9	I	外部输入 9	18	EGND		外部电源参考地

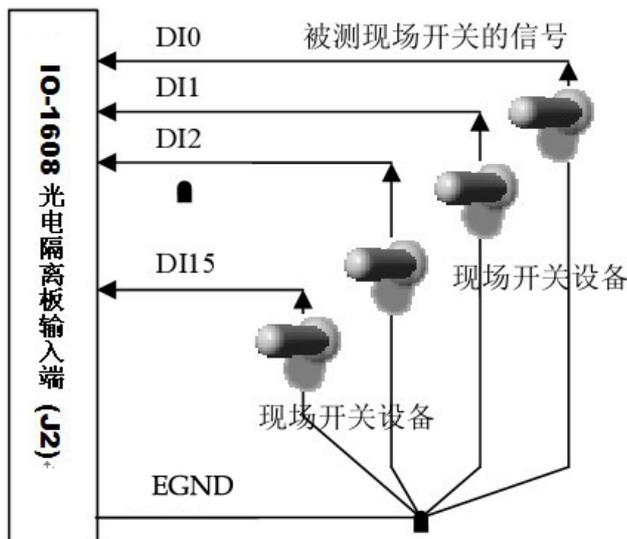


图 7_4 DI 数字信号量输入接线方式

7.2.3 J3---外部输出接口

引脚	名称	I/O	功能	引脚	名称	I/O	功能
1	OUT1	O	外部输出 1	6	OUT6	O	外部输出 6
2	OUT2	O	外部输出 2	7	OUT7	O	外部输出 7
3	OUT3	O	外部输出 3	8	OUT8	O	外部输出 8
4	OUT4	O	外部输出 4	9	EGND		外部电源参考地
5	OUT5	O	外部输出 5	10	EGND		外部电源参考地

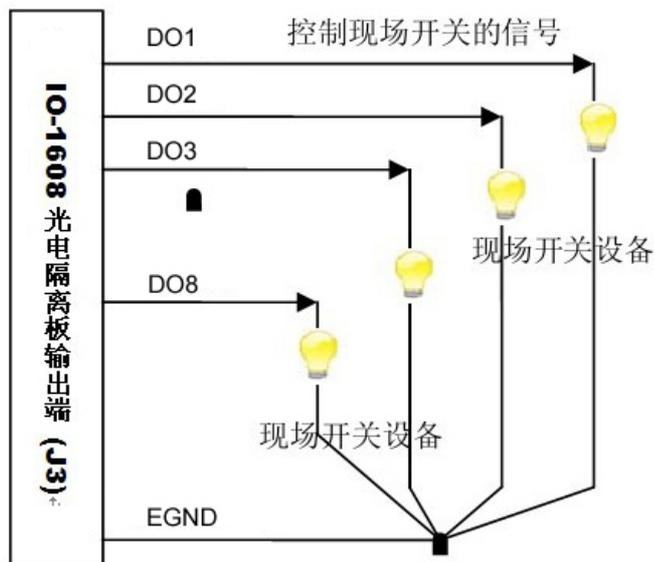


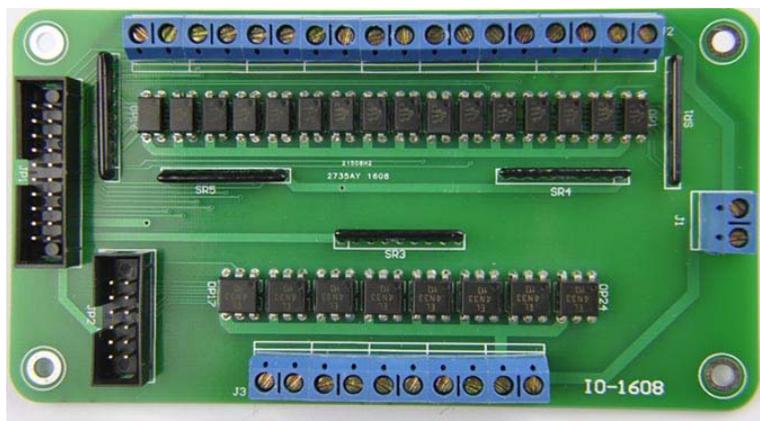
图 7_5 DO 数字信号量输出接线方式

7.2.4 ECC3910 采集卡与 IO-1608 光电隔离板连线对应表

ECC3910 采集卡		IO-1608 光电隔离板		连接电缆
接口	描述	对应接线接口	描述	描述
JP_IN	输入, 双列排针, 20 位	JP1	双列排座, 20 位	IDC20 扁平电缆, 长度 200mm
JP_OUT	输出, 双列排针, 12 位	JP2	双列排座, 12 位	IDC12 扁平电缆, 长度 200mm

- 注: 1. 接线时要注意引脚序号一一对应。
2. 连接电缆长度如果不满足要求, 可在订货前说明定制。

7.3 IO-1608 光电隔离板尺寸图



IO-1608 光电隔离板尺寸和 ECC3910 采集卡的尺寸一致, 可参考第八章。

八、ECC3910 采集卡机械尺寸图

